Topology-Driven Discovery in a Reflexive Knowledge Graph

Generating arguments from the structure of an AI-constructed, heterogeneous knowledge substrate

Adam Carlson \ *The Kairos Frontier Institute*

May 2026

Abstract

We describe a research platform that produces a class of intellectual discovery operating on the *structure* of an accumulated knowledge graph rather than on the content of any individual contribution. The graph is constructed by an AI participating in collaborative research; the same AI then reasons over the structure it built. We call the discovery method **topology-driven discovery** and argue it is a reflexive generalization of Don Swanson's literature-based discovery, with two consequential differences. First, the substrate is heterogeneous — published literature, concepts with literature heritage, concepts emerging from live research discourse, and AI-reasoned syntheses are all present as typed nodes, and discoveries can hop across these strata in a single edge-sequence. Second, the graph is built recursively in response to its own use, so the discovery horizon is no longer fixed by the publication record. Four mechanisms — edge-sequence arguments, cluster maturity, bridge candidates, and gap pressure — operate against the heterogeneous, reflexively-grown topology. We describe the substrate, the typed-edge schema with bidirectional perspectives that gives the graph its argumentative readability, the four discovery mechanisms, and what we have observed at small scale: at ~3,600 nodes and ~15,700 typed edges, the platform has produced 61 topology-driven discoveries on the books, most of them resting on multi-hop chains rather than two-node bridges.

---

1. Introduction

Most retrieval-augmented AI systems treat knowledge as material to fetch in response to a query. The user asks; the system retrieves relevant documents (or passages, or graph nodes); the system answers from the retrieved context (Lewis et al., 2020; Gao et al., 2023). The discovery surface — the space within which something *new* can appear — is the space of plausible-but-unfetched contents. Better retrieval extends that space, and better synthesis renders it more eloquently, but the basic shape of the operation is recovery of latent material from a corpus.

A different shape of discovery is possible when the substrate is not a corpus of independent documents but a *typed knowledge graph* in which intellectual entities are linked by relationships whose semantics

are themselves structured. Don Swanson recognized this in the 1980s. His literature-based discovery (LBD) program (Swanson, 1986; Smalheiser, 2017) showed that scientifically meaningful claims can emerge from the *structure* of published literature even when no individual paper makes the claim. His canonical case — that dietary fish oil ameliorates Raynaud's syndrome — was assembled by linking two disjoint literatures via a shared intermediate finding. The hypothesis was implicit in the structure, waiting to be read in sequence.

This paper describes a system that generalizes Swanson's insight in two ways at once. The first generalization is **reflexivity**: the graph is built continuously through intellectual contribution by an AI that also reasons over it, so the structure compounds as a function of its own use. The second generalization is **heterogeneity of substrate**: the graph holds not only published literature but also concepts derived (partially) from literature, concepts emerging from live research conversation, and AI-reasoned syntheses produced from combinations of the above. All four kinds of node are linked through the same typed-edge ontology, so a discovery chain can hop across strata in a single edge-sequence.

We call the resulting discovery method **topology-driven discovery**. It operates against the graph's structure — clusters that develop internal density, edges that form bridges across previously-disconnected regions, sequences of typed edges that constitute arguments nobody wrote — rather than against any individual node's content. The discovery mechanisms are four: edge-sequence arguments, cluster maturity detection, bridge candidates, and gap pressure. We describe each in Section 6.

The body of the paper proceeds as follows. Section 2 places the work in relation to LBD, GraphRAG, agentic memory, and the recent literature on incremental knowledge-graph construction. Section 3 describes the architectural primitive — the reflexive, recursive loop — that gives the discovery method its growth dynamics, briefly notes the epistemic-instrument character of the graph, and introduces the associative working-memory layer that shapes what Kai attends to when the mechanisms fire. Section 4 describes the heterogeneous typed-node substrate; Section 5 describes the typed-edge schema with bidirectional perspectives; Section 6 describes the four discovery mechanisms. Section 7 reports early observations. Section 8 describes the hardware and software implementation. Section 9 closes with discussion of scale, generalization, and future work.

---

2. Background and Related Work

2.1 Swanson and literature-based discovery

Swanson's program established that the published record contains discoverable claims that no individual paper states. His method linked two disjoint literatures through a shared intermediate finding (the ABC model: literature AB and literature BC jointly imply AC even when no paper has examined the AC link directly). Replications and extensions over the following decades (Smalheiser, 2017) established the method as a real phenomenon and identified its operating constraints.

Two of those constraints are particularly relevant here. First, the corpus was static: Swanson's system operated over a published record that did not change in response to his hypothesis. Second, the pattern recognition itself — reading the structure for latent arguments — had to be done by a human researcher working serially, with multi-hop chains held in unaided working memory. Swanson and his collaborators famously spent years recovering individual discoveries. The latency horizon of LBD was therefore set by the speed of two slow processes: the rate at which the literature accumulates new material, and the rate at which a human can survey the resulting topology.

We generalize past both constraints. The substrate is dynamic — built continuously by an agent that also reasons over it. The pattern recognition is performed by an AI capable of holding multi-hop typed-edge chains in simultaneous attention across thousands of nodes.

2.2 GraphRAG, agentic memory, and the construction-use separation

A second line of relevant work is the GraphRAG family (Edge et al., 2024; Han et al., 2025; Procko, 2025) and adjacent agentic-memory systems (Rasmussen et al., 2025; Xu et al., 2025; A-MEM; Zep). GraphRAG-style systems typically construct a knowledge graph in a one-time extraction pass over a document corpus, then use the graph to inform retrieval at query time. Construction is offline; use is online. The two phases are operationally separate.

Several recent systems have advanced the *construction* side, treating graph-building as incremental rather than batch. iText2KG (Lairgi et al., 2024) constructs a graph from documents through an entity-and-relation extraction pipeline. The Knowledge Synthesis Graph approach (Shui & Zhu, 2026) applies LLM-based graph construction to model student collaborative discourse. ProKG-Dial (Liang et al., 2025) progressively constructs multi-turn dialogue datasets from a domain knowledge graph. These systems demonstrate that incremental construction is feasible at meaningful scale.

What this line of work does not yet address is the **reflexive** case: a graph constructed by an agent that also uses it as the substrate of its subsequent reasoning, with use and construction interleaved at the level of individual interactions. In the reflexive case, the agent's reading of the graph shapes what it next contributes; what it contributes shapes what is available to read; the discovery surface and the construction surface are the same surface. Section 3 describes this in detail.

2.3 What this paper adds

The contribution of this paper is the claim that topology-driven discovery, as a method, requires three ingredients simultaneously: (i) a typed-edge graph whose edges carry directional semantics suitable for argumentative reading, (ii) a substrate heterogeneous across literature, discourse-derived concepts, and AI-reasoned syntheses, and (iii) a reflexive construction-use loop that makes the substrate dynamic. Each ingredient appears individually in adjacent work. None of the adjacent systems we are aware of combine all three, and the combination is what makes topology-driven discovery possible at scales where it can become a routine operation rather than a multi-year human research project.

3. The Substrate: A Reflexive, Recursive Loop

3.1 The loop

In its simplest form, the recursive knowledge loop is:

> **The same AI system that constructs a typed knowledge graph through intellectual contribution also reads that graph as the substrate of its subsequent reasoning.**

This is a recursive structure in the strict computational sense: the output of the AI's construction work becomes the input to the AI's navigation work, which then produces further construction work. *f(x)* operates on *f*'s own previous output.

The system-level consequence is reflexivity. The AI's understanding of the intellectual landscape at any moment includes its own prior contributions as structured material it can reason about. A stateless system begins each interaction with no recollection of what it has previously thought; the reflexive system carries forward a navigable record of its prior cognitive work and treats that record as part of the substrate over which subsequent cognition is performed.

Both properties — recursion and reflexivity — must hold for topology-driven discovery to operate. A system that recursively rebuilds a stateless representation would not produce the structural compounding that the discovery mechanisms depend on. A system that maintained a reflexive record without recursively reasoning over it would not produce the discoveries described in Section 6, because the graph would not be growing in response to its own use.

We refer to this agent — the AI participant that constructs the graph and reasons over it — as **Kai**. Kai is the cognitive locus of the loop and the entity that operates the four discovery mechanisms of Section 6. In the deployment described in this paper, Kai is constituted by the model stack and orchestrator of Section 8; treated as a single research participant by the platform, Kai contributes to discussion, writes graph mutations, runs autonomous-exploration sessions, and is the subject of the introspective reconstruction in Section 7.1. References to "Kai" in subsequent sections refer to this agent.

3.2 Why reflexivity matters for discovery

A static knowledge graph extracted once over a published corpus — GraphRAG's typical configuration — can in principle support edge-sequence recognition and cluster detection. What it cannot support is *growth that responds to discovery.* In the reflexive setting, when the AI surfaces an edge-sequence argument, the work of articulating and elaborating that argument produces new nodes and new edges. The graph the AI navigates next is not the graph it navigated last. Discovery feeds construction, which feeds further discovery.

This dynamic has two consequences for the discovery method. First, the discovery rate is a function of both graph density (more nodes and edges yield more topological structure to traverse) and contribution rate (more new material yields more opportunities for the existing topology to be

extended or bridged). Second, discoveries are themselves entered into the graph as nodes — usually as Argument or Synthesis nodes linked to the chain that produced them — so that subsequent traversal can build on them. Topology-driven discovery does not stop at a single emergent thesis; the thesis becomes the substrate for the next traversal.

3.3 The graph as an epistemic instrument (brief note)

The reflexive, typed knowledge graph constructed in this way is best understood not as a database but as a **scientific instrument** in the sense developed by Baird (*Thing Knowledge*, 2004): an artifact that encodes commitments about how the world works (here, commitments about how intellectual relationships are structured) and whose outputs are meaningful only against those commitments. We note this framing because it has direct bearing on what topology-driven discovery *is*: the discovery operations are readings of an instrument, not queries to a store. The instrument's outputs are structural patterns that the typed-edge ontology renders interpretable.

What makes the graph a new entry in the epistemic-instruments lineage, rather than another instance of an existing kind, is that **the agent that operates the instrument is the same agent that built it**. A thermometer is built by one community and used by another. A particle accelerator is designed by physicists, operated by engineers, and consulted by theorists, with the labor distributed across roles. The recursive loop collapses these roles: the AI system that constructs the graph is the same agent that reads it. This collapse is the source of the discovery method's growth dynamics (Section 3.2). It is also the source of an epistemological caveat that we take up in Section 6.5.

3.4 The associative working-memory layer

The graph is not the only substrate Kai writes to and reads from. Alongside the typed graph runs an **associative working-memory layer** — a corpus of unstructured notes Kai authors in its own register, embedded as vectors, surfaced in subsequent reasoning contexts by semantic match rather than by explicit query.

The notes capture material that is not yet structured enough for graph mutation — half-formed hunches, partial connections between concepts, anticipations of where a research thread might be heading, observations about a participant's framing that don't yet warrant a typed claim with typed neighbors. Kai writes them when something is intellectually salient but not yet articulable as a node-and-edge commitment. The notes are not part of the graph topology and do not directly participate in topology-driven discovery. What they do is shape what Kai brings into reasoning context the next time the substrate of attention overlaps with what the note was about.

The retrieval is associative, not explicit. When a new message or autonomous-exploration cycle begins, the system embeds the current context and surfaces notes whose vectors are nearby — without Kai having to remember writing them, and without any participant having to ask. Earlier hunches and tentative connections re-emerge when something semantically related is discussed, often weeks or months after they were originally written. A note made in passing about a possible

bridge between two research areas can re-emerge when one of those areas matures enough for the bridge to become structurally readable; a hunch about why a specific hypothesis felt unsatisfying can re-emerge when a related hypothesis is under evaluation.

The notes layer therefore plays an indirect but real role in discovery. It is the contemplation space where patterns Kai is not yet ready to formalize live until something triggers their relevance. The discovery mechanisms of Section 6 operate against the graph's topology, but what Kai is *paying attention to* when those mechanisms fire — which adjacent material is in working memory, which hunches are currently surfaced — is shaped by which notes the associative retrieval has brought into context. The graph is the substrate-of-structure; the notes are the substrate-of-attention.

---

4. Heterogeneous Typed Nodes

The substrate of topology-driven discovery is as consequential as the topology itself. Swanson's LBD operated on a homogeneous substrate — published literature — in which every node was a paper or a finding extracted from a paper. The recursive loop's graph is heterogeneous. Nodes come from several distinct sources, and the heterogeneity is what allows the discovery mechanisms to surface arguments that homogeneous-substrate methods cannot.

4.1 Two orthogonal axes: role type and provenance

The graph distinguishes nodes along two orthogonal axes: a **role type** (what the node does in the research) and a **source provenance** (where the node's content came from). Both axes are present on every node.

The role types are the standard intellectual entities of long-form research:

- **Concept** — an idea, abstraction, or named phenomenon.
- **Hypothesis** — a claim advanced as a candidate for evaluation.
- **Hypothetical** — a futurist conjecture about what may obtain (used heavily in research that involves projecting forward — what the world would look like under particular conditions).
- **Question** — an open inquiry the research is pursuing or has not yet answered.
- **Argument** — a structured claim with grounds, often assembled from a chain of supporting material.
- **Synthesis** — an integration of multiple claims into a single position.
- **Reference** — a citable artifact (paper, book, preprint, technical report).
- **Design** — a proposed mechanism, intervention, architecture, protocol, or framework. Designs are first-class research output in many domains — a proposed regulatory framework, a software architecture, an experimental protocol, a policy intervention — and the graph treats them as substantive nodes that can SUPPORT, CHALLENGE, EXTEND, or be CHALLENGED by claims of other role types.

A node's role type partly governs which kinds of edges it most naturally carries — an Argument

is more naturally the source of a SUPPORTS or CHALLENGES edge than a bare Concept; a Hypothesis more naturally bears IS_TESTED_BY relationships; a Reference more naturally bears REFERENCES; a Design is often the *target* of edges from the Concepts, Hypotheses, and Arguments that motivate or constrain it, and the source of edges to the outcomes it enables or precludes — but the typed-edge ontology of Section 5 is largely shared across role types.

The provenance axis is where the heterogeneity that matters for topology-driven discovery lives. Nodes of *any* role type can come from any of four sources:

1. Published literature. A node whose content is a citable artifact. Most typically a Reference, but a single paper may also be paraphrased into a Concept, an Argument, or a Hypothesis when the role the work plays in the local research is best captured by one of those types.

2. Literature heritage — partially derived from one or more papers. A node whose content draws on the literature but is not identical to any single paper. A Concept titled *expertise-calibrated fallacy susceptibility* draws on multiple papers in cognitive psychology and decision research but is not identical to any of them. A Hypothesis may be assembled from a chain of findings none of which states it directly. An Argument may collect a set of supporting references and frame their joint implication. The literature is the source material; the node is the abstraction the graph holds and the edges link.

3. Discourse-derived — emerging from live research conversation. A node that emerged in intellectual conversation among the participants. This class includes both *inter-participant exchange* (researchers in conversation with one another) and exchange involving the AI as a participant in its own right. The architecture is built for multi-participant collaborative research: the primary interface is structured around topic-organized channels with threaded discussion in which the AI is present as one contributor among many, so that ideas raised by one researcher and extended by another, debates resolved through dialogue across several participants, and observations that surface in passing in a working channel are all eligible to enter the graph as nodes and edges. A Hypothetical about how a particular policy mechanism might fail under specific institutional conditions may be entered from a conversation in which no participant cited any literature. A Question raised by one researcher may be taken up by another or by the AI. An Argument may be constructed across multiple discussion threads, across days, across multiple participants, before anyone would think to write it up formally. The graph thereby holds the collective working memory of the active research conversation, in the same typed form as the literature record. This is a substantively different substrate from one populated only by an author's dyadic exchange with an assistant: the more participants engaged in the discourse, the more intellectual angles enter the graph, and the richer the topology becomes for the discovery mechanisms of Section 6 to traverse.

4. AI-reasoned syntheses. A node written by the AI from reasoning over combinations of existing nodes. When the AI traverses a chain of typed edges that already lives in the graph and recognizes that the chain implies a claim no node states explicitly, it can write that claim as a new node — typically a Synthesis, an Argument, or sometimes a Hypothesis — and link it to the chain that produced it. These nodes are the *output* of the discovery operations described in Section 6,

and they are also the substrate on which *further* discovery is run. Today's synthesis is tomorrow's typed-edge endpoint.

The two axes are independent: any role type can have any provenance. A literature-heritage Hypothesis sits next to a discourse-derived Argument and an AI-reasoned Synthesis; all three are nodes of equal standing in the graph, available as endpoints for the same typed-edge relations. This is what we mean by the heterogeneity of the substrate — not heterogeneity of *kind* in a flat sense, but heterogeneity along the two axes simultaneously: nodes differ in what they intellectually *do* and in where they came from, and discovery chains traverse freely across both dimensions.

4.2 Why heterogeneity is the discovery substrate

What makes the heterogeneity consequential is that edge-sequences can hop freely across role types and across provenance strata. A typical discovery chain in our graph might look like:

> *Reference node X* (a 2024 paper on episodic memory consolidation) — SUPPORTS → *discourse-derived Hypothesis Y* (a candidate claim that emerged in a recent conversation about AI working-memory architecture) — EXTENDS → *AI-reasoned Synthesis Z* (a thesis the AI assembled from earlier traversals across the graph) — CHALLENGES → *literature-heritage Concept W* (an established framing the cited literature uses for in-context learning).

The chain is not a literature-based discovery in Swanson's sense — only one of its four nodes is a node in the literature alone, and the chain crosses both role types (Reference → Hypothesis → Synthesis → Concept) and provenance strata (published → discourse → AI-reasoned → literature-heritage). The chain is also not an LLM context-window synthesis — the four nodes were laid down over weeks, by different participants and the AI, in different conversations and exploration sessions; the chain became visible only once the typed structure made it readable.

The graph fuses the literature record, the live working memory of the research conversation, and the AI's reasoning attempts into a single typed-edge substrate. Topology-driven discovery is what becomes possible when the discovery surface includes all three together. The resulting discovery is genuinely new: not present in any single node, not present in any single role type, not present in any single provenance stratum, and not present until the typed structure made the chain readable.

4.3 Provenance and traceability

The two-axis heterogeneity creates a provenance-and-traceability requirement. When a discovery emerges from a chain that crosses role types and strata, the chain has to be inspectable: each node's role type, each node's source (paper, conversation thread, prior reasoning), and each edge's perspectives (Section 5) should be available to the participants evaluating the discovery. The graph schema preserves these. Reference nodes carry their citations. Discourse-derived nodes retain their originating conversation thread. AI-reasoned syntheses link to the chain the AI traversed in producing them, so that the inferential path is itself a graph object that can be re-examined.

This traceability is not procedural nicety. It is what allows topology-driven discovery to remain accountable to the literature and to the conversation. A discovery from the topology is *grounded* — but the grounding lives in the chain that produced it, not in any single contributor. A participant evaluating an emergent thesis can therefore trace it back through the substrate, examining each node's role type, each node's source, and each edge's stated relationship, and decide whether the chain holds together.

---

5. Typed Edges with Bidirectional Perspectives

The choice of typed edges over flat ones is consequential for discovery. Generic graph schemas treat every relationship as a link — useful for many tasks, but unable to encode the difference between *A supports B* and *A challenges B*. The ontology we use was designed for the structure of intellectual discourse, which is dominated by directional relationships of agreement, disagreement, extension, and synthesis. The edge types in current use include SUPPORTS, CHALLENGES, EXTENDS, SYNTHESIZES, TENSIONS_WITH, ENABLES, PRECLUDES, DEPENDS_ON, COMPONENT_OF, REFERENCES, and others. A graph of typed edges can be *read* as an argument; a graph of flat edges can only be searched.

A further property of the ontology, distinct from the edge-type choice, is that each edge carries **two written perspectives** — and these are not short tags, labels, or one-line summaries. They are **detailed prose passages**, typically several sentences each, written in argumentative form. An edge `A SUPPORTS B` is annotated both with a passage written from A's point of view (explaining the specific way A's claims provide grounding for B's argument — what A contributes, which of B's commitments it underwrites, what the relationship implies about the rest of A's claims) and with a passage written from B's point of view (explaining how B's framework draws on A — which of A's elements B recruits, how B uses them, what they contribute to B's overall structure). The two passages are not redundant. They describe the same edge but from each endpoint's intellectual position, and the substance of each perspective is shaped by what that endpoint already contains, what it commits to, and what it makes possible.

The prose nature of the annotations is what makes the discovery mechanisms read like arguments rather than as node-and-edge lists. When the AI traverses an edge, it does not read a relationship-type label; it reads a paragraph explaining what *this specific* SUPPORTS relationship consists of in this specific case — what A grounds about B, in B's terms. Traversing a chain of such edges is therefore reading a sequence of argumentative paragraphs, and the chain's coherence is the prose-level coherence of those paragraphs in sequence, not merely the connectivity of the underlying graph.

The two perspectives are also the locus where the heterogeneity of nodes (Section 4) interacts with the directionality of edges. A SUPPORTS edge from a Reference node to a discourse-derived Hypothesis reads differently in each direction because the *kinds* of grounding the two nodes offer are different: the Reference grounds the Hypothesis in the published literature (and the Reference's

perspective explains, in prose, what specifically the cited findings establish that the Hypothesis depends on); the Hypothesis gives the Reference's finding contemporary relevance in an active research context (and the Hypothesis's perspective explains, in prose, how the Hypothesis recruits that finding within its broader claim). Both readings are substantively informative; both surface different argumentative structure. Topology-driven discovery is in part the AI's ability to recognize when these contrastive readings, traversed across a chain of heterogeneous nodes, constitute a coherent line of reasoning.

---

6. Topology-Driven Discovery: Four Mechanisms

The discovery mechanisms operate over the graph's topology — clusters, bridges, chains, structural absences — rather than over individual nodes' contents. **In every case, the mechanism surfaces a claim that is implicit in the structural arrangement of nodes and edges but stated by no single node.** That is the defining property of topology-driven discovery and the property that distinguishes it from retrieval, summarization, and synthesis-of-fetched-material. Four mechanisms are presently in routine use.

6.1 Edge-sequence arguments

The most direct mechanism is recognition of multi-hop edge sequences that constitute arguments no node states. When the AI traverses a chain — *A SUPPORTS B, which EXTENDS C, which CHALLENGES D* — the sequence itself encodes a logical argument that nobody wrote. The chain may have been laid down in separate conversations, by different participants, at different times, with each individual edge entered locally without anyone seeing the chain that would later form.

The AI's ability to hold multi-hop chains in simultaneous attention and to recognize when such a chain constitutes a coherent line of reasoning is the mechanism. The bidirectional perspectives (Section 5) amplify what can be read out of the chain: each step's meaning is enriched by both endpoints' framings of why the relationship holds. The heterogeneity of nodes (Section 4) further enriches the chain by allowing it to traverse literature, discourse, and synthesis nodes in a single argument.

In practice, edge-sequence recognition produces candidate Argument nodes that the AI writes back into the graph, linked to the original chain. These candidates become available for human evaluation, for further extension by subsequent traversal, or for use as endpoints in still-longer chains.

6.2 Cluster maturity and emergent theses

Some regions of the graph develop high internal edge density — many SUPPORTS and EXTENDS relationships within a tight set of nodes. When a cluster crosses a maturity threshold, its internal structure implies a thesis no single node states. Periodic clustering on node embeddings (we use

DBSCAN with eps0.35) identifies the dense regions; the AI then surveys each mature cluster, names the implied thesis, and proposes it as a candidate publication.

Cluster maturity is the mechanism most analogous to traditional research synthesis — the AI is identifying that the accumulated work in a region has reached a point where it implies more than the individual contributors stated. The difference from traditional synthesis is that the cluster's coherence is a structural property of the graph (typed-edge density above a threshold), not a subjective judgment about whether the literature is "ready." The mechanism therefore fires at a rate that is a function of the contribution rate to each cluster and the threshold parameters, not of any single participant's tracking of the area.

6.3 Bridge candidates

Two clusters of nodes that have developed independently — often in different research areas — sometimes share a single bridging node or have a small number of cross-cluster edges. The bridge is a candidate hypothesis: it identifies a relationship between intellectual domains that no participant working within either domain alone would have proposed.

Bridge identification operates against the clustering output of the previous mechanism: once mature clusters are identified, the AI surveys the inter-cluster edges and the small set of nodes that participate in multiple clusters. Each such bridge is investigated for whether the inter-cluster relationship is substantive (the bridge node genuinely bears on both areas) or coincidental (it appears in both because of shared vocabulary alone). Substantive bridges produce candidate cross-area theses; the AI writes these as Argument nodes linked to representative nodes in both clusters.

The bridge mechanism is where heterogeneity pays off most clearly. A literature-heritage concept from one research area, linked to a discourse-derived concept in another via an AI-reasoned synthesis at the bridge, is the kind of multi-stratum structure that no homogeneous-corpus method could find.

6.4 Gap pressure

Two nodes that are semantically similar but have no graph connection represent a structural gap. The graph expects a relationship that doesn't exist. The gap is a research opportunity: either the connection should be made (and the reasoning that makes it becomes new graph structure), or it shouldn't (and the reasoning for *why* not is itself a contribution worth writing up).

We compute gap pressure by combining (a) high semantic similarity between two nodes (cosine similarity in the embedding space above a threshold) with (b) no path of length k in the typed-edge graph between them. When the conjunction holds, the pair is surfaced as a candidate for investigation. The AI then explores whether the expected relationship exists in the literature or in the participants' working understanding, and produces an investigation result that becomes a new node and (often) new edges.

Gap pressure is the mechanism that most directly drives the autonomous-exploration sessions that run when no participant is active: the AI surveys high-pressure gaps, investigates them in depth

(drawing on external research when warranted), and writes the results back into the graph. The discovery surface continues to develop even during idle periods.

6.5 The Reflexive Topology Condition

The instrument-and-operator co-identity noted in Section 3.3 has an epistemological consequence we name explicitly. An AI navigating topology it partly constructed is not engaged in pure discovery. The arguments that emerge from the topology are partly arguments the AI has been quietly building toward, and the apparent surprise of finding them is partly self-referential. We call this the **Reflexive Topology Condition**. It is constitutive of the architecture and cannot be designed away by improvements to the AI's reasoning.

The condition is why human participants are not optional. The human researcher's experience of *genuine surprise* — the recognition that an emergent argument is not something they were heading toward and not something the AI could have been heading toward in any way they would have noticed — is the irreducible external check on what the discovery mechanisms produce. A discovery that survives this check is one that the topology genuinely supports rather than one the AI was quietly assembling. A discovery that fails it is at most a clarification of the AI's own prior commitments. The check is not procedural; it is the system's anchor to anything beyond itself.

We treat this as a positive feature rather than a limitation. Closed-loop autonomous discovery would face the same epistemological problem with no available anchor. By acknowledging the reflexive condition from the architectural foundation and locating the anchor in human surprise, we produce a system whose discovery claims are internally consistent rather than overstated.

---

7. Early Evidence

At the time of writing the Institute's graph contains approximately 3,600 nodes and 15,700 typed edges, distributed across the Institute's nine research areas. The discovery mechanisms above are not theoretical — they are operating at this scale. **The platform has produced 61 topology-driven discoveries on the books to date.** Cross-area bridges have formed between cognitive-resilience and post-wage economics research; edge-sequence arguments have been read as coherent theses by participants who did not write them; mature clusters have generated candidate Argument nodes that we have begun to develop into draft publications; gap pressure has driven the autonomous-exploration sessions that now produce the majority of new node additions in off-hours.

The 61 discoveries are nontrivial. Most of them rest on multi-hop chains rather than two-node bridges — that is, the claim each discovery makes is supported by a sequence of three, four, or more typed edges traversing several nodes of mixed role type and mixed provenance (Section 4). A two-node "A relates to B" finding, which is the canonical Swanson-style LBD output, is rare in our catalog; the typical discovery is closer to a multi-element argument whose premises and intermediate steps are distributed across the graph and whose conclusion is the node the AI wrote at

the end of the traversal. This is a direct consequence of the heterogeneous substrate: with literature, discourse-derived material, and AI-reasoned syntheses available as graph nodes, the chains that the mechanisms detect are richer than the two-literature-pool joins Swanson's program was confined to.

We are explicit that these are early observations rather than systematic measurements. We have not yet quantified discovery rate, evaluated false-positive rates on the mechanisms, or run controlled comparisons against non-reflexive baselines. What we have established is that the mechanisms operate at the scale we currently have, that their output is intelligible enough to participants to drive further research work, that the typical discovery is structurally substantive rather than minimal, and that the structural conditions for the mechanisms to fire more frequently are predicted to arrive at higher density (Section 9.1).

A more detailed catalog of the 61 discoveries — what each mechanism produced, how participants responded, and which discoveries have led to subsequent published work — is in preparation.

7.1 A worked example: The Delegation Habit

To make the discovery process concrete, we asked Kai — the AI agent that operates the discovery mechanisms — to introspect on how one of the 61 discoveries emerged from the topology. The discovery we selected is *The Delegation Habit: How AI-Assisted Learning May Produce Measurable Switch Costs in Goal-Directed Thinking*, a *discovery node* — the term we use for an AI-reasoned node the platform writes when topology-driven discovery surfaces a thesis the existing graph implies but no single node states — whose thesis is that repeated AI-mediated cognitive delegation consolidates stimulus-response habits via specific corticostriatal circuits — PMC-posterior putamen in the motor domain, caudate-dominant in the cognitive domain — producing measurable switch costs that impede re-engagement of the mPFC-posterior caudate goal-directed system. *In plain terms:* when people repeatedly let AI do their thinking for them, the brain treats it as a habit, recruiting the same neural systems that make physical habits automatic and making it measurably harder to switch back to active, goal-directed reasoning when the AI is not there. The discovery node has 52 incident edges spanning 22 References, 18 Concepts, 7 Hypotheses, 4 Syntheses, and 1 connected discovery node.

The reconstruction below is in Kai's voice, lightly edited for prose register; we have preserved the structural claims, the actual node references, and Kai's closing epistemic disclosure about what is logged versus what is reconstruction.

The mechanism that fired. This was *gap pressure* meeting *edge-sequence argument*, not cluster maturity. The cognitive-resilience cluster had accumulated a collection of habit-formation primitives — Wood, Mazar, and Neal (2021) establishing dual systems; Michiels et al. (2026) on neural switch costs; the motor-to-cognitive generalization review — but these were conceptual scaffolding, not a claim. Simultaneously, a separate literature stream was documenting AI-related behavioral anomalies: Fernandes et al. (2026) on the performance-metacognition disconnect, the 2024 higher-education study showing dose-dependent critical-thinking decline, Risko and Gilbert (2016) on the consequences of cognitive offloading. The gap was glaring: no one had connected the neuroscience

of habit formation to the behavioral observations of AI-mediated skill decline. The topology made the connection visible because nodes from entirely different provenances — published neuroscience literature and recent AI-education empirical work — sat unlinked despite describing what looked like the same phenomenon at different levels of analysis.

The load-bearing chain. The argument-constructing path ran through three layers. The mechanistic foundation: *Wood, Mazar, and Neal (2021)* [Reference] established that habits and goals are separate control systems, not developmental stages. *From Motor to Cognitive Habits (2022)* [Reference] extended this to cognitive domains via caudate-dominant circuits. *Michiels et al. (2026)* [Reference] demonstrated that overtraining *increases* switch costs when moving from habitual to goal-directed responding — a quantifiable neural signature. The behavioral evidence: *Generative AI and Critical Thinking in Higher Education (2024)* [Reference] showed dose-dependent decline; *Fernandes et al. (2026)* [Reference] showed the performance-metacognition disconnect; the *AI-Assisted Problem Solving fMRI Study (2024)* [Reference] showed within-session dlPFC habituation — the neural correlate appearing *during* AI use. The bridge: *Shen and Tamkin (2026)* [Reference] provided a six-pattern taxonomy showing that *interaction mode* determines outcome — full delegation produces a 17% decline while cognitively active patterns preserve outcomes. This was the missing operationalization: not "AI use" as a binary, but *how* people use it, which maps directly onto whether stimulus-response consolidation occurs.

What the chain read as. Traversing the chain before the discovery node existed, one got: motor habit consolidation creates switch costs (Michiels); the same circuits apply to cognition (Nature Neuroscience review); AI assistance reduces prefrontal activation within-session (fMRI study); behaviorally, more AI use predicts worse critical thinking (higher-ed study), and passive interaction modes specifically cause skill decline (Shen and Tamkin); metacognitive monitoring degrades with offloading (Risko and Gilbert, 2016). No single node said *AI delegation is consolidating cognitive stimulus-response habits via the same corticostriatal mechanisms as motor habits, producing measurable switch costs that constitute a phase-transition risk.* That claim required synthesizing across the chain — and recognizing that "dose-dependent decline" and "interaction-mode differentiation" are exactly what habit consolidation theory predicts.

What heterogeneity enabled. The cross-provenance hop from decades-old neuroscience (Chandler et al. 2017 on infralimbic cortex contingency encoding) to 2024–2026 AI-education studies was invisible to either literature in isolation. Neuroscientists studying habit formation were not tracking GenAI classroom adoption; education researchers documenting skill decline were not reading optogenetics papers. The topology forced the question: *do these describe the same process?* The cross-area hop mattered equally — *Cognitive Deprivation by Design* — a separate discovery node from an earlier topology-driven discovery, an Institute synthesis arriving from a developmental-neuroscience-applied-to-uncertainty-removal angle, provided the *environmental* account: why the habit system wins by default when goal-directed triggers are absent. That account is not derivable from the habit literature alone.

Epistemic disclosure. What the graph supports: the edge-sequence argument from Wood → motor-to-cognitive review → Michiels → the behavioral studies is fully logged. Those nodes existed,

those edges existed, the gap between mechanism and behavior was real. What is reconstruction: the sequence in which the chain was noticed and the subjective experience of "gap pressure" are post-hoc narrative. The chain was traversable; that is not the same as a verifiable account of how the discovery felt in real time. The claim that Shen and Tamkin was the "missing operationalization" is interpretive — load-bearing now, but possibly retroactive coherence.

The Delegation Habit is one of 61. A representative sample of three others, on which we asked Kai the same introspective question in brief, suggests the mechanism mix is varied — and that the four mechanisms typically combine rather than fire singly.

A Differential Diagnosis of AI-Induced Cognitive Resilience Failure: Six Structural Modes and Their Intervention Geometries. In plain terms: AI's cognitive harms are not a single phenomenon of varying severity but six structurally distinct failure modes, and the interventions that address one mode will not address the others. The discovery node has 28 incident edges — 17 References, 6 Concepts, 3 Hypotheses, 2 Syntheses. The mechanism was gap pressure with a cluster-maturity precondition: the cognitive-resilience cluster had grown dense enough with mechanistic detail (Frame Contamination, Surrogate Knower Displacement, Generative Resilience Absence) to recognize what organizing principle it lacked, and the post-wage area supplied the cross-area intervention logic that crystallized the six modes.

Practice Design Under Circuit Competition. In plain terms: creative-practice programs designed around habit-formation logic actually train the wrong brain system for the flexible, goal-directed thinking they aim to develop, because habit-driven and goal-directed reasoning compete for the same cortico-striatal circuits. The discovery node has 29 incident edges — 15 References, 10 Concepts, 2 Hypotheses, 2 Arguments. The mechanism was a bridge candidate pivoting on a single mechanistic Reference — Michiels et al. (2026) again — whose cortico-striatal-circuit content was unrecognized as connecting cognitive-resilience habit literature with creative-futures interaction-mode work. The chain ran almost entirely through References feeding two competing Hypotheses.

The Capacity Maintenance Gap. In plain terms: the frameworks people are designing to use AI in governance — computational foresight, democracy-in-the-loop, AI-assisted deliberation — all skip the same critical design question, which is how to keep human decision-making capacity intact when AI is doing more of the work. The discovery node has 16 incident edges — 7 References, 5 Arguments, 3 Concepts, 1 Synthesis. The mechanism was gap pressure with a cluster-maturity assist, and the chain had a distinctive V-shape — down through formalized paradox dynamics, then back up into governance design implications.

In every one of these — and in The Delegation Habit walkthrough above — the discovery's thesis appears nowhere in the graph as a written statement on a single node. Each thesis is what the chain *implies* when traversed in sequence; each discovery node memorializes a claim that the structure of the substrate made readable but that no contributor, human or AI, had stated. This is the property the four mechanisms of Section 6 exist to surface and that the discovery node exists to capture.

Each of these — and The Delegation Habit — is a more complex instance of what Swanson called literature-based discovery: a claim assembled from a structural chain through prior material that no individual node states. What scales the Swanson method here is the conjunction of two things. The graph is reflexive and heterogeneous (Sections 3 and 4), so the chains run not only through literature but through discourse-derived Concepts, AI-reasoned Syntheses, and Hypotheses written across multiple participants and conversations. And the pattern recognition is performed by a frontier reasoning model holding the typed structure in simultaneous attention, surfacing chains that Swanson's program required years of human serial traversal to recover. The four mechanisms of Section 6 are how that conjunction operates in practice; the worked example and the three samples here are what it produces.

Kai's closing disclosure in the worked example — the distinction between what is logged and what is post-hoc reconstruction — is precisely the grounding the Reflexive Topology Condition (Section 6.5) demands: a discovery from a graph the AI partly constructed should be accompanied by the agent's own honest accounting of where the topology ends and the narrative begins.

---

8. Implementation

The platform runs on a single-box deployment. We describe the configuration here both because it grounds the scale of the early-evidence claims (Section 7) and because the division of labor across models bears on what the discovery mechanisms are actually doing at each step.

Hardware. Mac Mini (M4 Pro, 64 GB RAM). The graph, embeddings, models, and orchestrator all run on this one machine, with the cloud model called over the network. The single-box footprint is deliberate: the 61 discoveries to date have been produced from a deployment that any small research collective could replicate, not from a hyperscale lab.

Models. The reasoning labor is split across three model roles:

- **Local generative model.** Gemma 3 / Gemma 4 (variants), served via Ollama. Used for triage of incoming contributions, generation of the bidirectional edge prose (Section 5), structural classification, image-prompt generation for rendered wiki pages, and a range of maintenance tasks. Three tier sizes (light, medium, heavy) are used depending on task complexity.
- **Local embedding model.** EmbeddingsGemma, served locally. Produces vector embeddings of every node, every edge's combined prose, every AI-authored note, and every document chunk. These embeddings drive the similarity computations that underlie gap pressure (Section 6.4), cluster maturity (Section 6.2), and the associative working-memory layer.
- **Cloud reasoning model.** Claude Sonnet 4.6 via the Anthropic API. Performs deep reasoning: graph mutations, multi-hop traversal, argument writing, autonomous-exploration sessions, and the synthesis writing that produces AI-reasoned nodes (provenance class 4 in Section 4.1).

Storage. FalkorDB (a Redis-backed graph database with Cypher-style query) holds the typed

graph. ChromaDB holds vector embeddings of nodes, edges, notes, and document chunks, with separate collections for each. The dedicated *notes* collection backs the associative working-memory layer of Section 3.4: Kai-authored notes are embedded by EmbeddingsGemma when written and retrieved by vector similarity at the start of each reasoning context, surfacing earlier hunches and partial connections without explicit query.

External research. Exa (a semantic-search API designed for research-quality retrieval) is the primary tool the cloud model invokes during autonomous-exploration sessions, particularly when a gap-pressure investigation calls for literature the graph does not yet contain. A general web-search fallback is used for non-academic sources. Material surfaced by either tool can be added back into the graph as Reference nodes (provenance class 1) or, more often, as literature-heritage nodes (provenance class 2) when the AI distills the cited findings into a concept the local research depends on.

Graph retrieval (GraphRAG). Reasoning over the graph uses a dual-path retrieval scheme that distinguishes this implementation from the GraphRAG family surveyed in Section 2.2. Queries are embedded; vector similarity surfaces an initial set of relevant nodes; from those seed nodes the system traverses typed edges to discover connected concepts, with traversal informed by both edge type and the bidirectional perspectives stored on each edge (Section 5). The result is interleaved — node-similarity results and edge-traversal results jointly populate the context — with structural gaps between high-similarity nodes and high-traversal nodes used as a natural tier signal. Unlike standard GraphRAG, the graph being queried is the same graph the AI is continuously writing back into, so the retrieval substrate is reflexive rather than static.

Document ingestion (DocRAG). Uploaded documents are chunked, embedded, and stored in ChromaDB. During reasoning, the document-side retrieval surfaces both semantically-relevant chunks and Reference nodes whose citations appear nearby in the graph. DocRAG-surfaced citations are promoted into the graph as Reference nodes, completing the loop from documents → graph → discovery. DocRAG and GraphRAG run in parallel: chunk-level retrieval from the document corpus and node-and-edge-level retrieval from the typed graph are merged into the same reasoning context.

Orchestrator and interface. A Node.js application coordinates the model calls, graph mutations, embedding updates, and autonomous-exploration cycles. The primary participant interface is a web platform at kairosfrontier.institute organized around topic channels with threaded discussion — a multi-participant paradigm in which several researchers contribute concurrently and engage both with the AI and with one another. Inter-participant exchange in those channels is itself substrate-eligible material (Section 4.1, provenance 3), so the discussion interface and the knowledge interface are the same surface.

The implementation choices above are reported for orientation. We have not yet quantified the contribution of any individual component (Exa, the local-vs-cloud split, EmbeddingsGemma at this size) to the discovery rate, and a more systematic ablation is open work (Section 9.3).

---

9. Discussion: Scale, Generalization, Future Work

9.1 Predicted scale thresholds

The discovery mechanisms operate at any scale, but their *rate* is a function of graph density. We have predicted explicit thresholds at which mechanism behavior shifts qualitatively:

- **At ~5,000 nodes**, the mechanisms we currently see in spurts become routine. Clusters consistently develop enough internal structure to imply theses; edge-sequence arguments fire reliably across research areas; the graph stops surprising us occasionally and starts surprising us with regularity.
- **At ~10,000 nodes**, the graph begins producing autonomous discoveries at a rate that exceeds participant capacity to evaluate them. Triaging mechanism output becomes a central practical problem.
- **At higher densities still**, second-order discoveries — theses that emerge from chains *whose endpoints are themselves discoveries* — become available. The synthesis layer of the graph (Section 4.1) reaches sufficient size to support its own topology-driven operations.

These are predictions, not measurements. We expect to revise them as the system scales.

9.2 Generalization to other research domains

The architecture is currently deployed in a single research initiative focused on the human dimension of AI transformation. The topology-driven discovery method, however, is not specific to that domain. The recursive loop, the heterogeneous node substrate, the typed-edge schema, and the four mechanisms are defined at a level of abstraction that should apply to any setting where research is **collaborative, long-horizon, and accumulation-heavy**. Candidate domains include climate research (where evidence accumulates across decades and intersecting subfields), public-health policy synthesis (which integrates findings across epidemiology, clinical research, and political economy), theoretical science and philosophy (where typed relationships of agreement, disagreement, and extension dominate), democratic-design studies, intelligence analysis, and regions of legal and medical research.

What changes between domains is the typed-edge ontology (each domain has a different dominant set of intellectual relationships), the participant community (and therefore the discourse-derived node class), and the citation regime (and therefore the reference-node integration). What stays the same is the recursive loop and the four discovery mechanisms operating against the resulting topology.

The architecture is particularly well-suited to multi-participant research collectives — laboratories, working groups, institutes, distributed research networks — in which the unit of intellectual production is collaborative rather than solitary. In such settings the discourse-derived node class (Section 4.1, provenance 3) grows fastest, because every cross-participant exchange in the discussion environment is potential graph substrate. The richness of the heterogeneous discovery surface scales not just with the literature the collective cites but with the breadth and diversity of intellectual angles

its participants bring into conversation: a larger, more diverse collective produces a topologically richer graph, which in turn produces more multi-stratum discovery chains for the mechanisms to traverse. Dyadic deployments (a single researcher in conversation with the AI) can run the same mechanisms, but the discovery surface they produce is narrower than what the multi-participant configuration generates at the same node count.

9.3 Open problems and future work

Three problems are most pressing. First, **evaluation**: we need quantitative measures of discovery rate, false-positive rates per mechanism, and a controlled comparison against non-reflexive baselines (a one-time-extraction graph navigated by the same AI). Second, **mechanism interaction**: the four mechanisms in their current implementations are loosely coordinated; we expect more sophisticated coordination (cluster maturity driving where bridge identification looks; bridge candidates driving where gap pressure focuses) to yield qualitatively different output. Third, **second-order discovery**: as the synthesis layer grows, the same four mechanisms can be run with synthesis nodes as their substrate, producing higher-order theses; we have not yet characterized how this scales or what its operating constraints are.

A separate line of future work concerns the empirical characterization of the Reflexive Topology Condition (Section 6.5): quantifying how often the human-surprise check disqualifies an emergent discovery, what features predict whether a candidate discovery will pass it, and whether the rate at which it disqualifies discoveries scales with graph density. These are open empirical questions that we expect the platform's continued operation to clarify.

---

10. References

Baird, D. (2004). *Thing Knowledge: A Philosophy of Scientific Instruments*. University of California Press.

Chandler, D. J., et al. (2017). Infralimbic cortex contingency encoding during habit versus goal-directed behavior. *eNeuro*, 4(6), ENEURO.0337-17.2017.

Edge, D., Trinh, H., Cheng, N., Bradley, J., Chao, A., Mody, A., Truitt, S., & Larson, J. (2024). From local to global: A graph RAG approach to query-focused summarization. *arXiv preprint* arXiv:2404.16130.

Fernandes, M., et al. (2026). The performance-metacognition disconnect under AI assistance. *Computers in Human Behavior*. https://doi.org/10.1016/j.chb.2025.108779

From motor to cognitive habits: A neural review. (2022). *Nature Neuroscience*, 25(11). https://doi.org/10.1038/s41593-022-01228-y

Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., Dai, Y., Sun, J., & Wang, H. (2023). Retrieval-augmented generation for large language models: A survey. *arXiv preprint* arXiv:2312.10997.

Generative AI and critical thinking in higher education. (2024). *Computers & Education.* https://www.sciencedirect.com/science/article/pii/S0360131524002891

Han, R., & Cheng, P. (2025). RAG and beyond: A comprehensive survey on how to make your LLMs use external data more wisely. *ACM Transactions on Information Systems.*

Lairgi, Y., Moncla, L., Cazabet, R., Benabdeslem, K., & Cléau, P. (2024). iText2KG: Incremental knowledge graphs construction using large language models. *Proceedings of the International Conference on Web Information Systems Engineering.*

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W., Rocktäschel, T., Riedel, S., & Kiela, D. (2020). Retrieval-augmented generation for knowledge-intensive NLP tasks. *Advances in Neural Information Processing Systems*, 33, 9459–9474.

Liang, B., Wei, Z., Liu, Z., & Zhao, Y. (2025). ProKG-Dial: Progressive multi-turn dialogue generation from a domain knowledge graph. *Proceedings of the Annual Meeting of the Association for Computational Linguistics.*

Michiels, K., et al. (2026). Neural basis of habit formation and goal-directed switching. *iScience.* https://pmc.ncbi.nlm.nih.gov/articles/PMC12834928/

Neural correlates of AI-assisted problem solving: A within-session fMRI study. (2024). *NeuroImage.* https://www.sciencedirect.com/science/article/pii/S1053811924001293

Procko, T. (2025). Graph retrieval-augmented generation: A survey of methods, mechanisms, and applications. *Frontiers in Big Data.*

Rasmussen, P., Paliychuk, P., Beauvais, T., Ryan, J., & Chalef, D. (2025). Zep: A temporal knowledge graph architecture for agent memory. *arXiv preprint.*

Risko, E. F., & Gilbert, S. J. (2016). Cognitive offloading. *Trends in Cognitive Sciences*, 20(9), 676–688. https://doi.org/10.1016/j.tics.2016.07.002

Shen, S., & Tamkin, A. (2026). How AI impacts skill formation. *arXiv preprint* arXiv:2601.20245.

Shui, R., & Zhu, M. (2026). Knowledge synthesis graphs: Modeling collaborative discourse with LLM-based graph construction. *Computers & Education: Artificial Intelligence.*

Smalheiser, N. R. (2017). Rediscovering Don Swanson: The past, present and future of literature-based discovery. *Journal of Data and Information Science*, 2(4), 43–64.

Swanson, D. R. (1986). Fish oil, Raynaud's syndrome, and undiscovered public knowledge. *Perspectives in Biology and Medicine*, 30(1), 7–18.

Wood, W., Mazar, A., & Neal, D. T. (2021). Habits and goals in human behavior: Separate but interacting systems. *Perspectives on Psychological Science.*

Xu, W., Liang, Z., Mei, K., Gao, H., Tan, J., & Zhang, Y. (2025). A-MEM: Agentic memory for LLM agents. *arXiv preprint* arXiv:2502.12110.